REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
01-04-2011	Proceedings	MAR 2011 - APR 2011

4. TITLE AND SUBTITLE

A New Perspective on GMM Subspace Compensation Based on PPCA and Wiener Filtering

5a. CONTRACT NUMBER
FA8720-05-C-0002

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Alan McCree, Doug Sturim, and Doug Reynolds

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02420

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

NSA
9800 Savage Rd
Ft. Meade, MD 20755

10. SPONSOR/MONITOR'S ACRONYM(S)

NSA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We present a new perspective on the subspace compensation techniques that currently dominate the field of speaker recognition using Gaussian Mixture Models (GMMs). Rather than the traditional factor analysis approach, we use Gaussian modeling in the sufficient statistic supervector space combined with Probabilistic Principal Component Analysis (PPCA) within-class and shared across class covariance matrices to derive a family of training and testing algorithms. Key to this analysis is the use of two noise terms for each speech cut: a random channel offset and a length dependent observation noise. Using the Wiener filtering perspective, formulas for optimal train and test algorithms for Joint Factor Analysis (JFA) are simple to derive. In addition, we can show that an alternative form of Wiener filtering results in the i-vector approach. thus tying together these two disparate techniques.

15. SUBJECT TERMS

speaker recognition, Gaussian mixture model, Wiener filter, probabilistic principal components analysis (PPCA), factor analysis

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Zach Sweet
a. REPORT	b. ABSTRACT	c. THIS PAGE	SAR	4	19b. TELEPHONE NUMBER *(include area code)*
U	U	U			781-981-5997

NS-55000

A New Perspective on GMM Subspace Compensation Based on PPCA and Wiener Filtering

Alan McCree, Doug Sturim, and Doug Reynolds

MIT Lincoln Laboratory
Lexington, MA 02420
[mccree, sturim, dar]@ll.mit.edu

Abstract

We present a new perspective on the subspace compensation techniques that currently dominate the field of speaker recognition using Gaussian Mixture Models (GMMs). Rather than the traditional factor analysis approach, we use Gaussian modeling in the sufficient statistic supervector space combined with Probabilistic Principal Component Analysis (PPCA) within-class and shared across class covariance matrices to derive a family of training and testing algorithms. Key to this analysis is the use of two noise terms for each speech cut: a random channel offset and a length dependent observation noise. Using the Wiener filtering perspective, formulas for optimal train and test algorithms for Joint Factor Analysis (JFA) are simple to derive. In addition, we can show that an alternative form of Wiener filtering results in the i-vector approach, thus tying together these two disparate techniques.

Index Terms: speaker recognition, Gaussian mixture model, Wiener filter, probabilistic principal components analysis (PPCA), factor analysis

1. Introduction

Modeling speakers with GMMs and then generating test cut scores by evaluating the likelihood of each possible speaker has long been a successful method in speaker recognition [1]. In the last few years, subspace methods have been shown to provide both convenient models for channel compensation as well as rapid speaker enrollment, particularly with the JFA approach [2]. More recently, the subspace parameters themselves, referred to as i-vectors, have been used for recognition [3].

In this paper we present an alternative perspective on these algorithms based on sufficent statistic scoring, Gaussian observation and channel noises, PPCA covariance modeling, and Wiener filtering. The structure of the paper is as follows. First, in Section 2 we present GMM scoring using sufficient statistics and introduce the supervector observation and channel noises. A Gaussian model for the channel noise results in a simple Gaussian likelihood evaluation in the model supervector space; the use of a structured covariance matrix with PPCA simplifies the evaluation formulas. Section 2.3 then introduces the concept of Wiener filtering in the supervector space, and shows that this leads to straighforward derivations of the JFA train and test formulas. In Section 3, we show that reversing the order of the Wiener filter and removing the observation noise rather than the channel noise results in i-vector approaches. Section 4 provides experimental results comparing these various approaches on the NIST SRE10 evaluation. Finally, concluding remarks are provided in Section 5.

This work was sponsored by the Department of Defense under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

2. GMM Sufficient Statistics: the Path to Supervectors

We begin with a review of GMM model training and testing procedures presented in terms of Gaussian supervector sufficient statistics and Wiener filtering. We start with the assumption that all models differ only in the mean parameters. To be more specific, we assume that each speaker can be represented by a GMM with speaker-specific means but shared weights and covariance matrices.

2.1. Sufficient Statistic Scoring

Traditionally, the likelihood of a set of vectors under a GMM model is evaluated by directly computing the likelihoods for each frame and multiplying them together, a process which we refer to as frame-by-frame scoring. However, it is well known, and the basis for maximum likelihood training of the parameters of a GMM via the EM algorithm, that this likelihood can equivalently be evaluated by first computing the sufficient statistics of the input vectors and then using a single formula for the total likelihood. We refer to this as sufficient statistic scoring. For each Gaussian, the statistics needed are the counts, sum, and sum of squares of the vectors assigned to that Gaussian; in addition there is an overall statistic related to the mixture counts. Note that for a GMM (unlike a single Gaussian), these statistics are model-specific, since the model parameters are needed to generate the alignment of input vectors to Gaussians.

In general, sufficient statistic scoring gives the same answer as frame-by-frame scoring but provides no computational advantage, so it is not used in the testing process. Its primary use is in the theory of model training, since it provides formulas for the derivatives necessary to find optimal GMM parameters. However, it can also be very helpful for testing in the particular instance that the alignment is already given (for example from the UBM rather than model-dependent) the likelihood ratio will be computed between two GMMs that only differ in means, with the same weights and variances. In this case, only one set of sufficient statistics is needed for all models, and only the counts and sum (or equivalently sample mean) are needed. We note that for a straightforward GMM-UBM speaker recognition system, this assumption of UBM-alignment gives a small performance degradation (on the order of 10%) as compared to using the correct per-model statistics. However, the use of a single set of sufficient statistics is critical for computational feasibility of the more sophisticated techniques to be described next.

The evaluation of GMM likelihood for a set of vectors then reduces to a single Gaussian evaluation in a vector space of all the Gaussian means stacked together, which we refer to as a *supervector*. This Gaussian likelihood for model i is given by

$$p(\bar{\mathbf{x}}|S_i) \sim N(\mathbf{m}_i, \boldsymbol{\Sigma}_n) \quad (1)$$

where $\bar{\mathbf{x}}$ is the test sample mean supervector, $\mathbf{m}_i$ is the model mean supervector, and $\boldsymbol{\Sigma}_n$ is an *observation noise* that shrinks

to zero as the number of vectors increases. More specifically, if the GMM covariances are diagonal then they can also be stacked into a supervector Σ_0 and the observation noise is a diagonal covariance matrix in the supervector space where each diagonal element is the corresponding covariance from Σ_0 divided by the count for this Gaussian.

The corresponding log-likelihood can be written as

$$\log p(\bar{\mathbf{x}}|S_i) = (\bar{\mathbf{x}} - \mathbf{m}_i)^T \Sigma_n^{-1} (\bar{\mathbf{x}} - \mathbf{m}_i) + C_0 \quad (2)$$

where C_0 is a constant which is the same for all models and can be ignored.

2.2. The Additive Noise Model

This observation noise is equivalent to an additive Gaussian noise, since under this model we observe

$$\bar{\mathbf{x}} = \mathbf{m}_i + \mathbf{n} \quad (3)$$

where $\mathbf{n}$ is Gaussian with zero mean and covariance Σ_n. This leads naturally to the question: what if the observed supervector is also corrupted by a channel (or session) noise? For example, suppose that the feature vectors are log filterbank energies, and the test sequence is from a new channel with unknown frequency response resulting in an additive offset to the log filterbanks. In this case we can write

$$\bar{\mathbf{x}} = \mathbf{m}_i + \mathbf{c} + \mathbf{n} \quad (4)$$

where $\mathbf{c}$ is an unknown offset (additive noise) due to the channel for this recording. If we assume that $\mathbf{c}$ is Gaussian with zero mean and covariance Σ_c then this is a straightforward mathematical problem. Since both noise terms are Gaussian, the corresponding likelihood for speaker i is also Gaussian and is given by

$$p(\bar{\mathbf{x}}|S_i) \sim N(\mathbf{m}_i, \Sigma_c + \Sigma_n) \quad (5)$$

We refer to this technique as full Gaussian scoring.

To estimate Σ_c, we can use a large training set of model variations between well-trained models and test cuts and compute a sample covariance matrix. This is often referred to as the *within-class covariance*. Without additional constraints, however, this covariance matrix will be extremely large (the square of the supervector dimension). For a 2048 mixture Gaussian with 60 dimensional feature vectors, the supervector size is 122,280 so a full covariance matrix has more than 10^{10} parameters. It would be much simpler if Σ_c were diagonal, but this is not a realistic assumption. For example, in our hypothetical frequency response case $\mathbf{c}$ would be identical for each individual Gaussian in the supervector, so many components would be highly correlated. A straightforward approach to reducing the number of parameters in the channel covariance matrix is with Principal Component Analysis (PCA), where we keep only the eigenvectors of the covariance corresponding to the q largest eigenvalues. This is equivalent to assuming that the channel variation lies in a subspace of the supervectors. An even more powerful approach is Probabilistic PCA (PPCA) [4], in which the covariance matrix also includes a constant diagonal term so that it spans the entire space:

$$\Sigma_c = U_c U_c^T + \sigma^2 I. \quad (6)$$

If PCA or PPCA is used, the computation of the inverse of the covariance matrix needed to evaluate the likelihood of a model can be greatly simplified by the *matrix inversion lemma*. This formula requires only the inversion of a qxq matrix rather than a fullsize one:

$$(UU^T + D)^{-1} = D^{-1} - D^{-1}U(I + U^T D^{-1} U)^{-1} U^T D^{-1}. \quad (7)$$

2.3. Wiener Filtering for Channel Compensation

As an alternative to evaluating the likelihood of the test sequence under both observation and channel noise, it might be simpler to compensate for the channel noise with a preprocessing step. This is the vector space equivalent of a noise suppression algorithm for time signals. Minimizing the mean square error between the clean and compensated supervectors results in a matrix Wiener filter [5].

2.3.1. Speaker-Dependent Channel Compensation

Recall our modeling assumption of Eq. 4. The MMSE estimate of the channel-compensated supervector assuming the model mean $\mathbf{m}_i$ is known is given by the Wiener filter:

$$\hat{\mathbf{x}} = \Sigma_n (\Sigma_c + \Sigma_n)^{-1} (\bar{\mathbf{x}} - \mathbf{m}_i) + \mathbf{m}_i \quad (8)$$

Equivalently, we can first estimate the channel supervector and then subtract it from the input using:

$$\hat{\mathbf{c}}_i = \Sigma_c (\Sigma_c + \Sigma_n)^{-1} (\bar{\mathbf{x}} - \mathbf{m}_i) \quad (9)$$

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} - \hat{\mathbf{c}}_i. \quad (10)$$

In either case, we then evaluate the model likelihood assuming only observation noise with Eq. 5. If PCA or PPCA is used to model the channel covariance, the matrix inversion required for Wiener filtering is the same one needed for the full Gaussian scoring in the previous section, so the matrix inversion lemma can again be used to avoid a large matrix inversion.

2.3.2. Speaker-Independent Channel Compensation

Channel-compensated GMM scoring requires computing a new channel offset for each speaker. If many models are to be scored, it is tempting to reduce complexity by using the same offset for all models. The assumption of a model-independent channel offset also allows for the possibility of feature domain pre-processing of the input signal [6]. A simple approximation is commonly used for this, namely to assume the model is actually the UBM so that:

$$\hat{\mathbf{c}} = \Sigma_c (\Sigma_c + \Sigma_n)^{-1} (\bar{\mathbf{x}} - \mathbf{m}_0). \quad (11)$$

Unfortunately, this assumption of a single channel offset for all models does provide some performance degradation, Fortunately, another simplification referred to as *linear* (or inner product) scoring experimentally seems able to compensate for this loss [7]. This consists of approximating the Gaussian evaluation with only the linear term:

$$(\hat{\mathbf{x}} - \mathbf{m}_i)^T \Sigma_n^{-1} (\hat{\mathbf{x}} - \mathbf{m}_i) = -2\hat{\mathbf{x}}^T \Sigma_n^{-1} \mathbf{m}_i + C_1 \quad (12)$$

We would argue that a better approach to model-independent compensation would be to use the MMSE estimate of channel offset when the model is unknown, which is given by the following Wiener filter:

$$\hat{\mathbf{c}} = \Sigma_c (\Sigma_s + \Sigma_c + \Sigma_n)^{-1} (\bar{\mathbf{x}} - \mathbf{m}_0) \quad (13)$$

where we need to know the mean and covariance of the model means, $\mathbf{m}_0$ and Σ_s, which will be discussed in the next section. Unfortunately we have found that using this equation in speaker recognition does not work well; the reasons for this are not yet clear.

2.4. GMM Model Training by Wiener Filtering

We can also use this formalism to derive the optimal estimate of the model mean for a new speaker enrollment. We begin without channel distortion, in which case a speaker's training data can be modelled by a sample mean supervector corrupted by additive observation noise as given by Eq. 3. The MMSE estimate of the model mean can be attained by Wiener filtering to remove the observation noise n:

$$\mathbf{m}_i = \mathbf{\Sigma}_s(\mathbf{\Sigma}_s + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_0) + \mathbf{m}_0. \quad (14)$$

We assume that the mean of all model means $\mathbf{m}_0$ ("typical speaker") is given by the UBM. In more general terminology, the speaker covariance $\mathbf{\Sigma}_s$ is referred to as the *across-class* covariance matrix. Similarly to the within-class case, we estimate this covariance using a sample covariance of model means over a large training set, and we need to assume a structured covariance matrix to reduce the number of parameters. If we assume $\mathbf{\Sigma}_s$ is diagonal with the form of a constant times the GMM covariance $\mathbf{\Sigma}_0$, this corresponds exactly to relevance MAP adaptation of a model from the UBM [1]. If we use a PCA structure, this becomes eigenvoice modeling. The advantage of the eigenvoice approach is faster training with small amounts data, since all Gaussians are updated even if only some are seen in training. Unfortunately, when a large amount of data is available the eigenvoice approach does not converge to the correct model unless the subspace assumption is exactly correct. With PPCA, we get a more general representation of the extended MAP (EMAP) approach which combines fast adaptation speed with complete convergence [8]. Note that if we normalize all supervectors to the GMM covariance (by multiplying by $\mathbf{\Sigma}_0^{\frac{1}{2}}$) then the constant PPCA term corresponds to relevance MAP; we use this normalization in all of our experiments.

In the presence of channel noise, a single enrollment cut is again represented by Eq. 4. Now the MMSE estimate requires removing both channel and observation noise by Wiener filtering:

$$\mathbf{m}_i = \mathbf{\Sigma}_s(\mathbf{\Sigma}_s + \mathbf{\Sigma}_c + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_0) + \mathbf{m}_0. \quad (15)$$

which is equivalent to the two-stage process of channel bias estimation with Eq. 13 followed by mean estimation:

$$\mathbf{m}_i = \mathbf{\Sigma}_s(\mathbf{\Sigma}_s + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \hat{\mathbf{c}} - \mathbf{m}_0) + \mathbf{m}_0. \quad (16)$$

This can be interpreted as channel compensation followed by EMAP training. Note that the equation for estimating channel offset is different for training than it was for test, since here the actual model is not yet known resulting in an additional Gaussian uncertainty.

These are the equations for a single enrollment cut. The precise equations for multiple enrollments are complicated, but a common approximation is to perform channel compensation on each cut and then sum statistics for the final EMAP training. More precisely, though, the amount of channel compensation needed should be reduced as the number of cuts increases, since the channel will be averaged out automatically even without explicit compensation.

The combination of a PCA within-class (channel) covariance with a PCA or PPCA across-class covariance provides exactly the Joint Factor Analysis equations [2].

3. Reversing the Order: I-vectors

So far, we have used Wiener filtering in two ways: at test time to remove channel noise, and during training to remove both channel and observation noise. Here we explore an alternative possibility of reversing the order during testing: remove the observation noise first and then evaluate the likelihood of the channel noise.

Again we start with our modeling assumption of Eq. 4, but now we obtain the MMSE estimate of the observation noise-compensated supervector using a Wiener filter:

$$\hat{\mathbf{x}} = \mathbf{\Sigma}_c(\mathbf{\Sigma}_c + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_i) + \mathbf{m}_i \quad (17)$$

We then evaluate the model likelihood using only channel noise:

$$p(\bar{\mathbf{x}}|S_i) \sim N(\mathbf{m}_i, \mathbf{\Sigma}_c) \quad (18)$$

Note that this is equivalent to estimating the channel with Eq. 9 and then evaluating

$$p(\hat{\mathbf{c}}_i|S_i) \sim N(0, \mathbf{\Sigma}_c) \quad (19)$$

This equation is straightforward with a PPCA model for $\mathbf{\Sigma}_c$, but we can expand the key term of the log likelihood to gain additional insight:

$$\begin{aligned}\hat{\mathbf{c}}_i^T\mathbf{\Sigma}_c^{-1}\hat{\mathbf{c}}_i &= (\bar{\mathbf{x}} - \mathbf{m}_i)^T(\mathbf{\Sigma}_c + \mathbf{\Sigma}_n)^{-1}\mathbf{\Sigma}_c\mathbf{\Sigma}_c^{-1}\mathbf{\Sigma}_c \\ &\quad (\mathbf{\Sigma}_c + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_i)\end{aligned}$$

For a PCA channel covariance (equivalent to a PPCA covariance as $\sigma^2 \to 0$), $\mathbf{\Sigma}_c = U_c U_c^T$, and

$$\hat{\mathbf{c}}_i^T\mathbf{\Sigma}_c^{-1}\hat{\mathbf{c}}_i = \hat{\mathbf{z}}_i^T\hat{\mathbf{z}}_i \quad (20)$$

where $\hat{\mathbf{z}}_i$ is the low-dimensional component of $\hat{\mathbf{c}}_i$ before mapping back to the full supervector:

$$\hat{\mathbf{z}}_i = U_c^T(\mathbf{\Sigma}_c + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_i) \quad (21)$$

This shows that we can equivalently evaluate the likelihood of a speaker model with:

$$p(\hat{\mathbf{z}}_i|S_i) \sim N(0, I_{q_c}) \quad (22)$$

Therefore, Wiener filtering the observation noise rather than the channel noise results in the evaluation of the log-likelihood of the particular model as a simple inner product in the low-dimensional channel space. This q_c-dimensional vector is referred to as an *i-vector* [3], and in this case it could also be referred to as a *speaker-dependent channel factor*.

3.1. Model Independent I-vectors

In a fashion similar to the channel compensation approach, we can replace the model-dependent observation noise compensation Wiener filter with a model-independent one based on the UBM:

$$\hat{\mathbf{x}} = (\mathbf{\Sigma}_s + \mathbf{\Sigma}_c)(\mathbf{\Sigma}_s + \mathbf{\Sigma}_c + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_0) + \mathbf{m}_0 \quad (23)$$

We can simplify this notation by introducing the total covariance as the sum of the channel (within-class) and model (across-class) covariances: $\mathbf{\Sigma}_{tot} = \mathbf{\Sigma}_s + \mathbf{\Sigma}_c$.

If we use PCA modeling for both the channel and model covariances, and assume that both lie in the same subspace, then a similar limiting approach as above leads to the following expression for evaluating the likelihood of the test observation given a model:

$$p(\hat{\mathbf{z}}|S_i) \sim N(\mathbf{m}_i^z, \mathbf{\Sigma}_c^z) \quad (24)$$

where $\hat{\mathbf{z}}$ is the low-dimensional component of the mean-removed $\hat{\mathbf{x}}$ before mapping back to the full supervector:

$$\hat{\mathbf{z}} = U_{tot}^T(\mathbf{\Sigma}_{tot} + \mathbf{\Sigma}_n)^{-1}(\bar{\mathbf{x}} - \mathbf{m}_0) \quad (25)$$

and $\mathbf{m}_i^z$ and $\mathbf{\Sigma}_c^z$ represent the model mean and channel covariance in the subspace.

This result shows that Wiener filtering the observation noise in a model-independent fashion implies that the log-likelihood of a particular model is a simple Gaussian evaluation of the channel covariance in the low-dimensional total variability space. This q-dimensional vector is another example of an i-vector, in this case a *total factor*.

System	Male		Female	
	1c	8c	1c	8c
GMM-UBM	14.60	6.55	17.76	8.59
GMM-UBM zt	11.40	5.24	13.84	7.33
GMM stat	18.87	7.44	21.99	8.19
GMM stat zt	11.18	5.24	14.34	7.33
JFA full	3.68	0.95	4.56	1.30
JFA SI WF	7.1	2.0	8.2	2.3
JFA SI linear	3.64	0.48	4.56	1.14
FA SI linear	4.13	0.48	5.55	0.91
ivec SD	7.22	4.35	10.98	8.19
ivec SI	3.32	0.60	5.06	1.72
ivec cosine	3.31	0.62	4.94	1.72

Table 1: EER Performance for NIST SRE10 Extended Evaluation Telephone Data

System	Male		Female	
	1c	8c	1c	8c
GMM-UBM	0.970	0.832	0.957	0.968
GMM-UBM zt	0.952	0.933	0.962	0.870
GMM stat	0.971	0.873	0.963	0.922
GMM stat zt	0.964	0.946	0.974	0.887
JFA full	0.586	0.227	0.528	0.322
JFA SI WF	0.71	0.38	0.70	0.43
JFA SI linear	0.544	0.210	0.570	0.319
FA SI linear	0.522	0.268	0.557	0.328
ivec SD	0.917	0.695	0.934	0.811
ivec SI	0.488	0.257	0.627	0.410
ivec cosine	0.472	0.252	0.647	0.419

Table 2: Normalized minDCF Performance for NIST SRE10 Extended Evaluation Telephone Data

4. Experimental Results

We have compared the performance of some of the systems described in this paper on the NIST SRE 2010 extended evaluation task [9]. We used a modified version of the MITLL JFA submission [10], using a 512-mixture GMM based on 39-dimensional telephone-bandwidth cepstral features including deltas, with feature mean and variance normalization to mitigate channel effects. The background model and speaker covariance where trained on Switchboard II as well as SRE 2004, 2005, and 2006 telephone data. Channel covariance training used the same data except for Switchboard. The PPCA dimension for speaker space was 300, and for PCA dimension for channel noise was 100. For the i-vector approach, the total PCA dimension was 400.

The following systems were tested:

- GMM-UBM: straightforward GMM-UBM system with frame-by-frame scoring
- GMM-UBM zt: as above with ZT-norm
- GMM stat: GMM-UBM using UBM-aligned statistics for scoring
- GMM stat zt: as above with ZT-norm
- JFA full: PPCA speaker model, PCA channel, full Gaussian scoring, ZT-norm
- JFA SI WF: JFA with speaker-independent Wiener filtering
- JFA SI linear: JFA with speaker-independent channel compensation and linear scoring
- FA SI linear: as above but diagonal speaker model
- ivec SD: speaker-dependent i-vector system
- ivec SI: speaker-independent i-vector system
- ivec cosine: reference approach using cosine scoring [3].

From the EER and minimum DCF results in Tables 1 and 2, we make the following conclusions:

- Sufficient statistic scoring results in a slight performance degradation which is eliminated by ZT-norm.
- All forms of channel compensation provide drastic improvement with the exception of JFA SI WF and ivec SD.
- Fast speaker adaptation with EMAP provides modest gain (JFA vs. FA)
- Scoring observation noise only, channel noise only, or both all result in similar performance.

5. Conclusion

In this paper we have presented an alternative perspective on subspace modeling in GMM-based speaker recognition. Working with sufficient statistics, Gaussian PPCA speaker and channel models, and Wiener filtering provided a straightforward approach to deriving JFA algorithms. An alternative approach to Wiener filtering yielded the i-vector approach, showing that both JFA and i-vectors can be derived from a single formalism.

6. References

[1] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[2] P. Kenny, "Joint factor analysis of speaker and session variability: Theory and algorithms," Tech. Rep. CRIM-06/08-13, CRIM, 2005.

[3] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front end factor analysis for speaker verification," *to appear in IEEE Transactions on Audio, Speech and Language Processing*, 2010.

[4] M. Tipping and C. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Computation*, vol. 11, pp. 435, 1999.

[5] H. Andrews and B. Hunt, *Digital Image Restoration*, Prentice-Hall, 1977.

[6] F. Castaldo, D. Colibro, E. Dalmasso, P. Laface, and C. Vair, "Compensation of nuisance factors for speaker and language recognition," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 1969 –1978, Sept. 2007.

[7] Ondrej Glembek, Lukas Burget, Najim Dehak, Niko Brummer, and Patrick Kenny, "Comparison of scoring methods used in speaker recognition with joint factor analysis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 2009.

[8] E. Jon, D. Kim, and N. Kim, "Robust correlation estimation for EMAP-based speaker adaptation," *IEEE Signal Processing Letters*, vol. 8, pp. 184–186, June 2001.

[9] "The NIST year 2010 speaker recognition evaluation plan," http://www.itl.nist.gov/iad/mig/tests/sre/2010.

[10] D. Sturim, W. Campbell, N. Dehak, Z. Karam, A. McCree, D. Reynolds, F. Richardson, P. Torres-Carrasquillo, and S. Shum, "The MIT LL 2010 speaker recognition evluation system: Scalable language-independent speaker recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 2011.